TANGO
Device
Server

Kicker Kicker
User's Guide

Kicker Class

Revision: start - Author: vedder
Implemented in C++ - CVS repository: ESRF

Introduction:

Interface class for the booster extraction Kicker Ki and Ke.

Class Identification:

- **Contact :** at esrf.fr - vedder
- **Class Family :** PowerSupply (ESRF Specific)
- **Platform :** Unix Like
- **Bus :** Modbus

Class Inheritance:

- Tango::DeviceImpl
    - PowerSupply
        - Kicker

Properties:

Device Properties		
Property name	**Property type**	**Description**
ModbusDevice	Tango::DEV_STRING	The name of the modbus device to use for comunication with the PLC.
RegisterBaseAddress	Tango::DEV_LONG	The base address for the block of registers used for this kicker power supply.

Device Properties Default Values:

Property Name	Default Values
ModbusDevice	No default value
RegisterBaseAddress	No default value

There is no Class properties.

States:

States	
Names	**Descriptions**
ON	Power Supply is ON
OFF	Power Supply is OFF
STANDBY	Power Supply is STANDBY
FAULT	Fault detected on the Power Supply
ALARM	Alarm detected on the Power Supply
MOVING	When the device is going on, it execute a sequence. During this execution, the device state is moving.

Attributes:

Scalar Attributes			
Attribute name	**Data Type**	**R/W Type**	**Expert**
Current: The powersupply current setting in amps	DEV_DOUBLE	READ_WRITE	No
Voltage: The powersupply voltage in volts.	DEV_DOUBLE	READ_WRITE	No
CurrentSetPoint: The current set value as stored in the powersupply.	DEV_DOUBLE	READ	No
PulseNumber	DEV_USHORT	READ_WRITE	No

Commands:

More Details on commands....

Device Commands for Operator Level		
Command name	**Argument In**	**Argument Out**
Init	DEV_VOID	DEV_VOID
State	DEV_VOID	DEV_STATE
Status	DEV_VOID	CONST_DEV_STRING
On	DEV_VOID	DEV_VOID
Off	DEV_VOID	DEV_VOID
Reset	DEV_VOID	DEV_VOID
Standby	DEV_VOID	DEV_VOID
SetCurrentWithoutApply	DEV_DOUBLE	DEV_VOID

Device Commands for Expert Level Only		
Command name	**Argument In**	**Argument Out**
DBG_On	DEV_VOID	DEV_VOID
DBG_Off	DEV_VOID	DEV_VOID
DBG_Reset	DEV_VOID	DEV_VOID
DBG_Standby	DEV_VOID	DEV_VOID
DBG_Apply_setting	DEV_VOID	DEV_VOID

1 - Init

- **Description:** This commands re-initialise a device keeping the same network connection.
  After an Init command executed on a device, it is not necessary for client to re-connect to the device.
  This command first calls the device *delete_device()* method and then execute its *init_device()* method.
  For C++ device server, all the memory allocated in the *nit_device()* method must be freed in the *delete_device()* method.
  The language device desctructor automatically calls the *delete_device()* method.

- **Argin:**
  DEV_VOID : none.

- **Argout:**
  DEV_VOID : none.

- **Command allowed for:**

  - Tango::ON
  - Tango::OFF
  - Tango::STANDBY
  - Tango::FAULT
  - Tango::ALARM
  - Tango::MOVING

2 - State

- **Description:** This command gets the device state (stored in its *device_state* data member) and returns it to the caller.

- **Argin:**
  DEV_VOID : none.

- **Argout:**
  DEV_STATE : State Code

- **Command allowed for:**

  - Tango::ON
  - Tango::OFF
  - Tango::STANDBY
  - Tango::FAULT
  - Tango::ALARM
  - Tango::MOVING

3 - Status

- **Description:** This command gets the device status (stored in its *device_status* data member) and returns it to the caller.

- **Argin:**
  DEV_VOID : none.

- **Argout:**
  CONST_DEV_STRING : Status description

- **Command allowed for:**

  - Tango::ON
  - Tango::OFF
  - Tango::STANDBY
  - Tango::FAULT
  - Tango::ALARM
  - Tango::MOVING

4 - On

- **Description:** Switch powersupply ON.

- **Argin:**
  DEV_VOID :

- **Argout:**
  DEV_VOID :

- **Command allowed for:**

  - Tango::OFF
  - Tango::STANDBY
  - Tango::ALARM

5 - Off

- **Description:** Switch powersupply OFF.

- **Argin:**
  DEV_VOID :

- **Argout:**
  DEV_VOID :

- **Command allowed for:**

  - Tango::ON
  - Tango::STANDBY
  - Tango::ALARM

6 - Reset

- **Description:** Reset the powersupply to a well known state.

- **Argin:**
  DEV_VOID :

- **Argout:**
  DEV_VOID :

- **Command allowed for:**

  - Tango::ON
  - Tango::OFF
  - Tango::STANDBY
  - Tango::FAULT
  - Tango::ALARM

7 - Standby

- **Description:** Set the kicker to standby

- **Argin:**
  DEV_VOID :

- **Argout:**
  DEV_VOID :

- **Command allowed for:**

  - Tango::ON
  - Tango::OFF
  - Tango::ALARM

8 - DBG_On (for expert only)

- **Description:** Send a ON command to the PLC, without any timing or state control. This is a debug function.

- **Argin:**
  DEV_VOID :

- **Argout:**
  DEV_VOID :

- **Command allowed for:**

  - Tango::ON
  - Tango::OFF
  - Tango::STANDBY
  - Tango::FAULT
  - Tango::ALARM
  - Tango::MOVING

9 - DBG_Off (for expert only)

- **Description:** Send a OFF command to the PLC, without any timing or state control. This is a debug function.

- **Argin:**
  DEV_VOID :

- **Argout:**
  DEV_VOID :

- **Command allowed for:**

  - Tango::ON
  - Tango::OFF
  - Tango::STANDBY
  - Tango::FAULT
  - Tango::ALARM
  - Tango::MOVING

10 - DBG_Reset (for expert only)

- **Description:** Send a Reset command to the PLC, without any timing or state control. This is a debug function.

- **Argin:**
  DEV_VOID :

- **Argout:**
  DEV_VOID :

- **Command allowed for:**

  - Tango::ON
  - Tango::OFF
  - Tango::STANDBY
  - Tango::FAULT
  - Tango::ALARM
  - Tango::MOVING

11 - DBG_Standby (for expert only)

- **Description:** Send a Standby command to the PLC, without any timing or state control. This is a debug function.

- **Argin:**
  DEV_VOID :

- **Argout:**
  DEV_VOID :

- **Command allowed for:**

  - Tango::ON
  - Tango::OFF
  - Tango::STANDBY
  - Tango::FAULT
  - Tango::ALARM
  - Tango::MOVING

12 - DBG_Apply_setting (for expert only)

- **Description:** This commands sends the apply settings to the PLC (Bit6 set in Control_word1). It’s only use for debugging purposes and should not be used in normal situation.

- **Argin:**
  DEV_VOID :

- **Argout:**
  DEV_VOID :

- **Command allowed for:**

  - Tango::ON
  - Tango::OFF
  - Tango::STANDBY
  - Tango::FAULT
  - Tango::ALARM
  - Tango::MOVING

13 - SetCurrentWithoutApply

- **Description:** This command is here to handle synchronised set current on Kicker. usually when current is set, the apply command is called immediately after. Here we only set the current value, without sendind PLC apply command. The apply command should be done on KickerSynchroCommand device to validate values for ALL Kicker.

- **Argin:**
  DEV_DOUBLE : current set point.

- **Argout:**
  DEV_VOID :

- **Command allowed for:**

  - Tango::ON
  - Tango::OFF
  - Tango::STANDBY
  - Tango::FAULT
  - Tango::ALARM
  - Tango::MOVING